UNITED STATES DISTRICT COURT
EASTERN DISTRICT OF MICHIGAN
SOUTHERN DIVISION

LEAGUE OF WOMEN VOTERS
OF MICHIGAN, et al.,

          Plaintiffs,

v.

RUTH JOHNSON, in her official
capacity as Michigan
Secretary of State, et al.

          Defendants.

_____/

Case No. 2:17-cv-14148

Hon. Eric L. Clay
Hon. Denise Page Hood
Hon Gordon J. Quist

DECLARATION OF YAN LIU

I, Yan Liu, declare under penalty of perjury and pursuant to 28 U.S.C. § 1746, as follows:

1.   I am over 18 years of age and am competent to testify.

2.   I have personal knowledge of the matters set forth below.

3.   I am a Senior Research Programmer at the National Center for Supercomputer Applications at the University of Illinois.

4.   I received my Ph.D. in Informatics from the University of Illinois. I have a Masters in Computer Science from the University of Iowa, an M.E. in Computer Engineering from Wuhan University in Wuhan, China, and a B.S. in Computer Science from Wuhan University.

5.   I have published more than forty peer-reviewed research articles, primarily in computer and computational sciences and the interdisciplinary science domains of scientific computing, geographic information science, and operations research. My research has been supported by the National Science Foundation, NASA, and the U.S. Geological Survey.

6.   In my career, I regularly draft and review software source code written in multiple computer programming languages, including Java.

7.   I was retained by the Defendant Secretary of State in June of 2018 and was asked by the Defendant to review and prepare a report discussing the proposed expert report of Dr. Jowei Chen.

8.   Dr. Chen's report purports to rely on his comparisons of Michigan's Current Apportionment Plan (the "Enacted Plan") against sets of simulated redistricting plans drawn by a computer using a program created by Dr. Chen.

9.   Shortly after being retained by the Secretary, I requested that her counsel obtain and provide to me a copy of Dr. Chen's computer source code for the programs he identifies in his report.

10.   The Secretary's counsel initially provided to me three compiled code files, ".jar" files, which I understood to have been provided by Plaintiffs' counsel on June 12, 2018 and to constitute Dr. Chen's byte code.

11.   Byte code is not human readable. It is coding language that is understood by machines, but is not particularly useful in discerning the instructions a programmer has given to the computer. It is not possible to discern Dr. Chen's instructions to the computer from reviewing the byte code.

12.   I requested that the Secretary's counsel obtain and provide to me a copy of Dr. Chen's source code. "Source code" is written by a programmer in a human-readable programming language and is the source of the byte code. Review of the source code would precisely disclose the instructions Dr. Chen gave to the computer for it to draw his simulated redistricting plans.

13.   It is possible from reviewing the .jar files (the files with names that end in ".jar") to tell when those files were compiled from the source code. This is because the .jar files, which are generated by the Java compiler and contain "Java Class" byte code, preserve the last modified date of the byte code, which is the compiling date. The .jar files provided by the Plaintiffs' counsel to the Secretary on June 12, 2018, and subsequently provided to me, showed a compiling date of June 12, 2018 for the byte code. This means that the source code used to generate the byte code existed as of June 12, 2018. One cannot compile byte code without source code.

14.   I took a screenshot that shows the compiling date of these Java Class byte code files. This screenshot is attached to this Declaration as Exhibit A. To me, this means that Dr. Chen's source code still existed on June 12, 2018, when the byte code was compiled.

15.   Following up on my request for Dr. Chen's source code, in late June, 2018, I understand that the Secretary's counsel obtained from Plaintiffs' counsel three text files. The Secretary's counsel provided these text files to me for review. I found that these files did not contain source code, but contained decompiled byte code.

16.   Decompiled byte code is binary machine code that has been re-translated back into coding language. The re-translation process must guess at the original coding language and substitute values and terms to fill in its assumptions— it is thus highly imperfect. Though portions may be readable, most portions generally are not. Much is lost in translation.

17.   Byte code can be decompiled using any number of free tools available to the public online. Through trial and error and using the previously produced byte code file, I was able to identify the tool likely used by Dr. Chen in creating the decompiled byte code files given to the Secretary's counsel. The decompiled version of the byte code provided to the Secretary's counsel however, appeared to have been edited to remove error messages produced by that decompiling tool. I do not know who removed these error messages.

18.   I was surprised to receive decompiled byte code. As a research programmer who frequently collaborates with others in drafting code, I would not share decompiled byte code and I would not expect my collaborators to understand my decompiled byte code. When we share code to review or to collaborate in drafting a program, we always share the actual source code.

19.   I again requested Dr. Chen's source code files from the Secretary's counsel. The Secretary's counsel advised me that they were not able to obtain a copy from Plaintiffs' counsel before the deadline for my report. I thus included a statement in my report that I had not been able to review the source code, but that such review may help to identify flaws in Dr. Chen's methodology.

20.   I understand that the Secretary's counsel continued to seek a copy of Dr. Chen's source code.

21.   In August of 2018, the Secretary's counsel advised me that Plaintiffs' counsel in turn had advised that Dr. Chen did not save the final version of his source code, but that there was a "draft" version they could provide that was the "closest" version to the final source code Dr. Chen could locate.

22.   I understand that Plaintiffs' counsel provided to the Secretary's counsel a copy of Dr. Chen's "draft" source code. The Secretary's counsel provided this to me for further review.

23.   From the draft source code, it is apparent that Dr. Chen did not allow for his simulations to produce plans that would be similar to the Enacted Plan. Dr. Chen included, for example, a command that his State Senate and House program should only output those simulated plans that had fewer county breaks than the Enacted Plan. This assured that his simulations would never be like the Enacted Plan.

24.   Also from my review of the "draft" source code, it appears that Dr. Chen's program takes a long time to produce simulated plans. The draft source code file for State House simulation includes an instruction directing the computer to terminate the simulation if a plan is not produced after 40 days.

25.   In my attempts to run Dr. Chen's compiled byte code file and draft code and to produce a single plan for State Senate and State House, I was unable to do so even after 10 days with a relatively powerful personal computer.

26.   I understand that Dr. Chen testified at his deposition that he used Eclipse and other java code drafting programs to create his source code.

27.   Eclipse and other code drafting tools typically prompt a user to save all changes to their source code before exiting the tool. A user would have to affirmatively decide not to save changes for the changes not to be preserved in the ordinary course.

28.   It is programming 101 to save the source code file, and incredible that a programmer that has spent considerable time and energy in drafting code— especially code that will be used to generate thousands of maps over the course of months for use in litigation—would not save the final version of the source code that was compiled into the byte code.

29.   By comparing the decompiled byte code and the "draft" source code for each of Dr. Chen's three programs (i.e., his programs for the Michigan House, Michigan Senate, and Congress), I can tell that at least 10 functions ("Java class methods") present in the draft source code files were modified or deleted prior to compiling in each set. I cannot, however, tell in every instance which functions were deleted or changed in the decompiled byte code because, as explained above, decompiled byte code is very difficult or impossible to read in most instances.

30.   I have been advised that Dr. Chen stated at his deposition that the changes he made to his draft source code before compiling it were "cosmetic." I can, however, tell that the decompiled byte code dramatically differs with respect to the instructions for how the simulation process should seek to achieve compactness in both the State Senate and State House simulations. Dr. Chen also appears to have changed (compared to the "draft" code) the output limitations in his State House simulation concerning the permissible number of county breaks, and he appears to have changed the output limitations concerning the permissible number of MCD breaks in both the State Senate and State House Plans. These new instructions, not present in the draft source code provided by Plaintiffs, would have altered how the plans were drawn and thus cannot reasonably be characterized as being "cosmetic."

31.   Any one or even all of the multiple changed functions identified in my review could have a significant impact on the rendering of plans. So too could there be still other changed functions that I was unable to identify from the decompiled byte code. A function does not need to be long or complex to have a significant effect on the output. Without final source code, however, I am not able to determine the effect of the functions that Dr. Chen deleted, modified, or potentially added to his source code between the "draft" version supplied and the final version actually compiled and used.

I declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct.

_____   December 03, 2018
            Yan Liu   Dated

EXHIBIT A

```
$ls
Computer_Code_For_Congressional_Simulations_CONFIDENTIAL_INFORMATION.jar
Computer_Code_For_House_Simulations_CONFIDENTIAL_INFORMATION.jar
Computer_Code_For_Senate_Simulations_CONFIDENTIAL_INFORMATION.jar
$jar tvf Computer_Code_For_Congressional_Simulations_CONFIDENTIAL_INFORMATION.jar
    94 Tue Jun 12 16:07:30 CDT 2018 META-INF/MANIFEST.MF
 26311 Tue Jun 12 16:07:30 CDT 2018 Simulations/Michigan_Congressional_District_Simulations.class
$
$jar tvf Computer_Code_For_Senate_Simulations_CONFIDENTIAL_INFORMATION.jar
    84 Tue Jun 12 16:08:38 CDT 2018 META-INF/MANIFEST.MF
 20225 Tue Jun 12 16:08:38 CDT 2018 Simulations/Michigan_State_Senate_Simulations.class
$
$jar tvf Computer_Code_For_House_Simulations_CONFIDENTIAL_INFORMATION.jar
    83 Tue Jun 12 16:08:04 CDT 2018 META-INF/MANIFEST.MF
 21664 Tue Jun 12 16:08:04 CDT 2018 Simulations/Michigan_State_House_Simulations.class
```